

WHAT IS CLAIMED IS:

1. A method for enforcing a life cycle process in a source control system, comprising:
 - receiving a user-defined life cycle process having a plurality of states, each state having attributes;
 - receiving user-defined state transitions between said plurality of states;
 - providing a change state function for changing a current state associated with an object to a next state associated with said object, said change state function verifying compliance with said user-defined state transitions; and
 - providing version control for said object in said source control system.
2. The method according to claim 1, wherein said version control comprises:
 - providing a check-in function; and
 - providing a check-out function.
3. The method according to claim 1, wherein said attributes include a fallback state.
4. The method according to claim 1, further comprising:
 - receiving user-defined security for said user-defined state transitions.
5. The method according to claim 4, wherein said user-defined security includes electronic signatures.
6. The method according to claim 4, wherein said user-defined security includes which users have permission to make which state transitions.

7. The method according to claim 1, wherein said object is a control strategy for a process control system.
8. The method according to claim 7, wherein said attributes include whether said control strategy is loadable to a controller.
9. The method according to claim 1, wherein receiving said user-defined life cycle process having said plurality of states, each state having attributes is performed through a user interface having an editable table, said table having state names as rows and attributes as columns and having cells indicating values for said attributes.
10. The method according to claim 6, wherein receiving user-defined state transitions between said plurality of states is performed through a user interface having an editable table, said table having state names as rows and column and having cells indicating which users have permission to make which state transitions.
11. A computer readable medium having executable instructions stored thereon to perform a method of determining permissions for actions with an object based on a state of said object, said method comprising:
 - receiving a request to perform an action with said object;
 - determining whether said object has ever been checked-in to a source control system;
 - determining whether said object is currently checked-in;
 - retrieving a definition of said state of said object;
 - determining from said definition whether said action is permissible in said state; and
 - providing a permission status.

12. A computer readable medium having executable instructions stored thereon to perform a method of validating state transitions, said method comprising:
- receiving a request to make a state transition for an object from a user;
 - determining whether said object is checked-in;
 - determining whether said user has permission to make said state transition based on a user-defined state transition model;
 - permitting said state transition, if said user has permission; and
 - providing a state transition status.
13. A computer readable medium having executable instructions stored thereon to perform a method of validating a state transition, said method comprising:
- determining whether a next state in a state transition request from a user is allowed from a current state in said state transition request based on user-defined transition restrictions;
 - determining whether said user has permission to make said state transition based on user-defined transition restrictions; and
 - providing a state transition status.
14. The computer readable medium according to claim 13, further comprising:
- determining whether said state transition has a restricted signing requirement and, if so, verifying that said restricted signing requirement is met.
15. A computer readable medium having executable instructions stored thereon to perform a method of validating a state transition, said method comprising:
- determining whether said current state transition in a state transition request for an object from a user requires an electronic signature based on user-defined transition restrictions;

determining whether a previous state transition for said object required a previous electronic signature, if said current state transition requires a current electronic signature;

allowing said current state transition only if said previous electronic signature is different than said current electronic signature; and

providing a validation status.

16. A computer readable medium having executable instructions stored thereon to perform a method of determining a new state for an object version upon check-in, said method comprising:

determining whether said object is being checked-in for a first time;

retrieving a first fallback state for a first pre-defined state, if said object is being checked-in for said first time; and

providing said first fallback state, if said object is being checked-in for said first time.

17. The computer readable medium according to claim 16, comprising:

retrieving a current state for a current version of said object, if said object is not being checked-in for said first time;

retrieving a current fallback state for said current state of said object, if said object is not being checked-in for said first time; and

providing said current fallback state, if said object is not being checked-in for said first time.

18. A computer readable medium having executable instructions stored thereon to perform a method of processing the addition of a state, said method comprising:

receiving a definition of a new state from a user, said definition including a name and a fallback state;

determining whether said name is unique among existing state definitions;

validating said fallback state; and
adding said definition to a source control system, only if said name is unique
and said fallback state is valid.

19. The computer readable medium according to claim 18, wherein said
definition includes a restricted signing requirement and further comprising:
validating said restricted signing requirement; and
wherein said adding said definition to said source control system is
performed on an additional condition of whether said restricted signing
requirement is valid.

20. The computer readable medium according to claim 18, further comprising:
determining whether said user has a privilege to edit said definition; and
wherein said adding said definition to said source control system is
performed on an additional condition of whether said user has said privilege.

21. A computer readable medium having executable instructions stored thereon
to perform a method of processing the modification of a state, said method
comprising:

receiving a modified definition of a state from a user, said modified
definition including a name and a fallback state;
determining whether said name is unique among existing state definitions;
validating said fallback state; and
updating said modified definition in a source control system, only if said
name is unique and said fallback state is valid.

22. The computer readable medium according to claim 21, wherein said
definition includes a restricted signing requirement and further comprising:
validating said restricted signing requirement; and

wherein said updating said modified definition in said source control system is performed on an additional condition of whether said restricted signing requirement is valid.

23. The computer readable medium according to claim 21, further comprising:
determining whether said user has a privilege to edit said definition; and
wherein said updating said modified definition in said source control system is performed on an additional condition of whether said user has said privilege.

24. A computer readable medium having executable instructions stored thereon to perform a method of processing the deletion of a state, said method comprising:
receiving a request to delete a state definition for said state from a user;
determining whether said state definition is referenced by any other state definition in a source control system;
determining whether any objects in said source control system have a current state equal to said state;
deleting said state definition from said source control system, only if said state definition is not referenced by any other state definition in said source control system and no objects in said source control system have said current state equal to said state.

25. The computer readable medium according to claim 24, further comprising:
determining whether said user has a privilege to delete said definition; and
wherein said deleting said state definition from said source control system is performed on an additional condition of whether said user has said privilege.

26. A source control system for a process control system, comprising:
a processor;

a life cycle process component executable on said processor to enforce compliance with user-defined life cycle states;

a version control component executable on said processor to associate a version number with each object; and

a controller in communication with said processor via a network to be loaded with said objects to provide process control for a plurality of devices.

27. The system according to claim 26, further comprising:
another processor to back-up said server.

28. The system according to claim 26, further comprising:
a state configuration component executable on said processor to receive state information from a user for each state

29. The system according to claim 28, wherein said state information includes a state name and an indication of whether load to controller is allowed from that state.

30. The system according to claim 28, wherein said state information includes a fallback state.

31. The system according to claim 28, wherein said state information includes an indication of whether restricted signing is needed.

32. The system according to claim 28, wherein said state configuration component provides editing functions for said state information.

33. The system according to claim 26, further comprising:

a state transition component executable on said processor to receive state transition configuration requirements from a user.

34. The system according to claim 33, wherein said state transition configuration requirements include which users have permission to make particular state transitions.

35. The system according to claim 33, wherein said state transition configuration requirements include an indication of whether an electronic signature is needed to make particular state transitions.

36. The system according to claim 26, wherein said version control component provides check-in and check-out functions.

37. The system according to claim 26, further comprising:
a change qualification state component to process a qualification state transition request from a user.